ISSN (Online): 2349-7084
GLOBAL IMPACT FACTOR 0.238
DIIF 0.876

INTERNATIONAL JOURNAL OF COMPUTER ENGINEERING IN RESEARCH TRENDS
VOLUME 2, ISSUE 9, SEPTEMBER 2015, PP 525-530

Clustering high-dimensional data derived from Feature Selection Algorithm

[1]Mohammad Raziuddin ,[2]T. Venkata Ramana

[1]Assistant Professor, NMREC
[2] Professor & HOD, CSE, SLC's IET
raziuddin.5807@gmail.com, meetramana_12@yahoo.co.in

Abstract: Clustering high-dimensional data is the cluster analysis of data with anywhere from a few dozen to many thousands of dimensions. Such high-dimensional data spaces are often encountered in areas such as medicine, where DNA microarray technology can produce a large number of measurements at once, and the clustering of text documents, where, if a word-frequency vector is used, the number of dimensions equals the size. Feature selection is the process of identifying a subset of the most useful features that produces compatible results as the original entire set of features. A feature selection algorithm may be evaluated from both the efficiency and effectiveness points of view. While the efficiency concerns the time required to find a subset of features, the effectiveness is related to the quality of the subset of features. Based on these criteria, a FAST clustering-based feature Selection algorithm (FAST) is proposed and experimentally evaluated. Features in different clusters are relatively independent; the clustering-based strategy of FAST has a high probability of producing a subset of useful and independent features.

Keywords: high-dimensional data spaces, filter method, feature clustering, graph-based clustering, feature Selection algorithm (FAST).

1. INTRODUCTION

Feature selection is an important topic in data mining, especially for high dimensional datasets. Feature selection (also known as subset selection) is an effective way for reducing dimensionality, removing irrelevant data, increasing learning accuracy. Feature selection [14] can be divided into four types: the Embedded, Wrapper, Filter, Hybrid approaches.

Clustering high-dimensional data is the cluster analysis of data with anywhere from a few dozen to many thousands of dimensions. Such high-dimensional data spaces are often encountered in areas such as medicine, where DNA microarray technology can produce a large number of measurements at once, and the clustering of text documents, where, if a word-frequency vector is used, the number of dimensions equals the size. Eg of high-dimensional data are

Data on health status of patients can be high-dimensional (100+ measured/recorded parameters from blood analysis, immune system status, genetic background, nutrition, alcohol- tobacco-drug-consumption, operations, treatments, diagnosed diseases ...)

The Embedded methods incorporate feature selection as part of the training process and are usually specific to given learning algorithms. Decision Trees is the one example for embedded approach. Wrapper model approach uses the method of classification itself to measure the importance of features set, hence the feature selected depends on the classifier model used. Wrapper methods generally result in better performance than filter methods because the feature selection process is optimized for the classification algorithm to be used. However, wrapper methods are too expensive for large

dimensional database in terms of computational complexity and time since each feature set considered must be evaluated with the classifier algorithm used [11]. The filter approach actually precedes the actual classification process. The filter approach is independent of the learning algorithm, computationally simple fast and scalable [11].

With respect to the filter feature selection methods, the application of cluster analysis has been demonstrated to show the effectiveness of the features selected from the point of view of classification accuracy [14]. In cluster analysis, graph-theoretic methods have been well studied and used in many applications. The general graph theoretic clustering is simple: compute a neighborhood graph then delete any edge in the graph that is much longer/ shorter (according of instances, to some criterion) than its neighbors. The result is a forest and each tree in the forest represents a cluster. In our study, we apply graph theoretic clustering methods to features.

2. RELATED WORKS

Zheng Zhao and Huan Liu in "Searching for Interacting Features" propose to efficiently handle feature interaction to achieve efficient feature selection [13]. S.Swetha, A.Harpika in "A Novel Feature Subset Algorithm for High Dimensional Data", build up a novel algorithm that can capably and efficiently deal with both inappropriate and redundant characteristics, and get hold of a superior feature subset [14]. T.Jaga Priya Vathana, C.Saravanabhavan, Dr.J.Vellingiri in "A Survey on Feature Selection Algorithm for High Dimensional Data Using Fuzzy Logic" proposed fuzzy logic has focused on minimized redundant data set and improves the feature subset accuracy [15]. Manoranjan Dash, Huan Liub in "Consistency-based search in feature selection", focuses on inconsistency measure according to which a feature subset is inconsistent if there exist at least two instances with same feature values but with different class labels. We compare inconsistency measure with other measures and study different search strategies such as exhaustive, complete, heuristic and random search that can be applied to this measure [16]. Mr. M. Senthil Kumar, Ms. V. Latha Jyothi M.E in" A Fast Clustering Based Feature Subset Selection Using Affinity Propagation Algorithm"-Traditional approaches for clustering data are based on
metric similarities, i.e., nonnegative, symmetric, and satisfying the triangle inequality measures using graph based algorithm to replace this process a more recent approaches, like Affinity Propagation (AP) algorithm can be selected and also take input as general non metric similarities [1]. Priyanka M G in "Feature Subset Selection Algorithm over Multiple Dataset"- here a fast clustering based feature subset selection algorithm is used. The algorithm involves (i) removing irrelevant features, (ii) constructing clusters from the relevant features, and (iii) removing redundant features and selecting representative features. It is an effective way for reducing dimensionality. This FAST algorithm has advantages like efficiency and effectiveness. Efficiency concerns the time required to find a subset of features and effectiveness is related to the quality of the subset of features. It can be extended to use with multiple datasets [2]. Lei Yu, Huan Liu in" Efficient Feature Selection via Analysis of Relevance and Redundancy"- we show that feature relevance alone is insufficient for efficient feature selection of high-dimensional data. We define feature redundancy and propose to perform explicit redundancy analysis in feature selection. A new framework is introduced that decouples relevance analysis and redundancy analysis. We develop a correlation-based method for relevance and redundancy analysis, and conduct an empirical study of its efficiency and effectiveness comparing with representative methods [17]. Yanxia Zhang, Ali Luo, and Yongheng Zhao in" An automated classification algorithm for multi-wavelength data" we applied a kind of filter approach named Relief to select features from the multi-wavelength data. Then we put forward the naive Bayes classifier to classify the objects with the feature subsets and compare the results with and without feature selection, and those with and without adding weights to features. The result shows that the naive Bayes classifier based on Relief algorithms is robust and efficient to preselect AGN candidates [18]. N.Deepika, R.Saravana Kumar in" A Fast Clustering Based Flexible and Accurate Motif Detector Technique for

High Dimensional Data", present an algorithm that uses FLAME as a building block and can mine combinations of simple approximate motifs under relaxed constraints. The approach we take in FLAME explores the space of all possible models. In order to carry out this exploration in an efficient way, we first construct two suffix trees: a suffix tree on the actual data set that contains counts in each node (called the data suffix tree), and a suffix tree on the set of all possible model strings (called the model suffix tree).

3. PROBLEMS OVER CLUSTERING IN HIGH-DIMENSIONAL DATA

Four problems need to be overcome for clustering in high-dimensional data:[1]

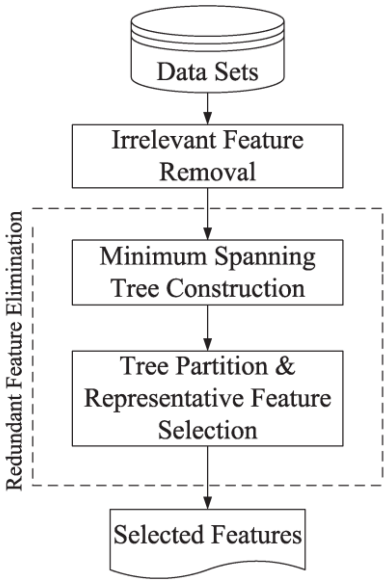
- Multiple dimensions are hard to think in, impossible to visualize, and, due to the exponential growth of the number of possible values with each dimension, complete enumeration of all subspaces becomes intractable with increasing dimensionality. This problem is known as the curse.
- The concept of distance becomes less precise as the number of dimensions grows, since the distance between any two points in a given dataset converges. The discrimination of the nearest and farthest point in particular becomes meaningless:

$$\lim_{d\to\infty} \frac{dist_{\max} - dist_{\min}}{dist_{\min}} = 0$$

- A cluster is intended to group objects that are related, based on observations of their attribute's values. However, given a large number of attributes some of the attributes will usually not be meaningful for a given cluster. For example, in newborn screening a cluster of samples might identify newborns that share similar blood values, which might lead to insights about the relevance of certain blood values for a disease. But for different diseases, different blood values might form a cluster, and other values might be uncorrelated. This is known as the local feature relevance problem: different clusters might be found in different subspaces, so a global filtering of attributes is not sufficient.
- Given a large number of attributes, it is likely that some attributes are correlated. Hence, clusters might exist in arbitrarily oriented affine subspaces.

4. FEATURE SELECTION

To remove irrelevant features and redundant features, the FAST [14] algorithm has two connected components. The irrelevant feature removal is straightforward once the right relevance measure is defined or selected, while the redundant feature elimination is a bit of sophisticated. In our proposed FAST algorithm, it involves 1) the construction of the minimum spanning tree from a weighted complete graph; 2) the partitioning of the MST into a forest with each tree representing a cluster; and 3) the selection of representative features from the clusters.



ISSN (Online): 2349-7084

INTERNATIONAL JOURNAL OF COMPUTER ENGINEERING IN RESEARCH TRENDS
VOLUME 2, ISSUE 9, SEPTEMBER 2015, PP 525-530

Fig 1.Clustering based FAST

A. Load Data

The data has to be pre-processed for removing missing values, noise and outliers. Then the given dataset must be converted into the raff format. From the raff format, only the attributes and the values are extracted and stored into the database. By considering the last column of the dataset as the class attribute and select the distinct class labels from that and classify the entire dataset with respect to class labels.

B. Entropy and Conditional Entropy Calculation

Relevant features have strong correlation with target concept so are always necessary for a best subset, while redundant features are not because their values are completely correlated with each other. Thus, notions of feature redundancy and feature relevance are normally in terms of feature correlation and feature-target concept correlation. To find the relevance of each attribute with the class label, Information gain is computed. This is also said to be Mutual Information measure.

Mutual information measures how much the distribution of the feature values and target classes differ from statistical independence. This is a nonlinear estimation of correlation between feature values or feature values and target classes. The Symmetric Uncertainty (SU) is derived from the mutual information by normalizing it to the entropies of feature values or feature values and target classes, and has been used to evaluate the goodness of features for classification. The SU is defined as follows:

$$SU(X,Y) = \frac{2 \times Gain(X|Y)}{H(X) + H(Y)}$$

Where, H(X) is the entropy of a random variable X. Gain (X|Y) is the amount by which the entropy of Y decreases. It reflects the additional information about Y provided by X and is called the information gain [13] which is given by

$$Gain(X|Y) = H(X) - H(X|Y)$$
$$= H(Y) - H(Y|X).$$

Where H (X|Y) is the conditional entropy which quantifies the remaining entropy (i.e., uncertainty) of a random variable X given that the value of another random variable
Y is known.

C. T-Relevance and F-Correlation Computation

The relevance between the feature Fi € F and the target concept C is referred to as the T-Relevance of Fi and C, and denoted by SU (Fi, C). If SU (Fi, C) is greater than a predetermined threshold, then Fi is a strong T-Relevance feature.

After finding the relevance value, the redundant attributes will be removed with respect to the threshold value. The correlation between any pair of features Fi and Fj (Fi, Fj € ^ F ^ i ≠ j) is called the F-Correlation of Fi and Fj, and denoted by SU (Fi, Fj). The equation symmetric uncertainty which is used for finding the relevance between the attribute and the class is again applied to find the similarity between two attributes with respect to each label.

D. MST Construction

With the F-Correlation value computed above, the MST is constructed. A MST [12] is a sub-graph of a weighted, connected and undirected graph. It is acyclic, connects all the nodes in the graph, and the sum of all of the weight of all of its edges is minimum. That is, there is no other spanning tree, or sub-graph which connects all the nodes and has a smaller sum. If the weights of all the edges are unique, then the MST is unique. The nodes in the tree will represent the samples, and the axis of the dimensional graph represents the n features.

The complete graph G reflects the correlations among all the target-relevant features. Unfortunately, graph G has k vertices and k(k-1)/2 edges. For high dimensional data, it is heavily dense and the edges with different weights are strongly interwoven. Moreover, the decomposition of complete graph is NP-hard. Thus for graph G,

build an MST, which connects all vertices such that the sum of the weights of the edges is the minimum, using the well known Kruskal's algorithm. The weight of edge (Fi`, Fj`) is F-Correlation SU(Fi`,Fj`).

Kruskal's algorithm is a greedy algorithm in graph theory that finds a MST for a connected weighted graph. This means it finds a subset of the edges that forms a tree that includes every vertex, where the total weight of all the edges in the tree is minimized. If the graph is not connected, then it finds a minimum spanning forest (a
MST for each connected component). If the graph is connected, the forest has a single component and forms a MST. In this tree, the vertices represent the relevance value and the edges represent the F-Correlation value.

E. Partitioning MST and Feature subset selection

After building the MST, in the third step, first remove the edges whose weights are smaller than both of the TRelevance SU (Fi`, C) and SU (Fj`, C), from the MST. After removing all the unnecessary edges, a forest F is obtained. Each tree Tj € F represents a cluster that is denoted as V (Tj), which is the vertex set of Tj as well. As illustrated above, the features in each cluster are redundant, so for each cluster V (Tj) chooses a representative features whose T-Relevance is the greatest. All representative features comprise the final feature subset.

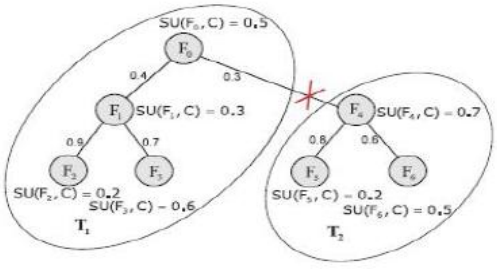


Fig. 2 Example of Minimum Spanning Tree

F. Classification

After selecting feature subset, classify selected subset using Probability-based Naïve Bayes Classifier with the help of Bayes concept.. Thus the naïve Bayes based classifier able to classify in many categories with the various label classification and feature selections from the output of the kruskal's where it generates the some filtered that MST values, which can formulates some cluster view with the help of the naïve Bayes concepts.

4. CONCLUSION

In this paper, we have proposed an Efficient FAST clustering-based feature subset selection algorithm for high dimensional data improves the efficiency of the time required to find a subset of features. The algorithm involves 1) removing irrelevant features, 2) constructing a minimum spanning tree from relative ones, and 3) partitioning the MST and selecting representative features. In the proposed algorithm, a cluster consists of features. Each cluster is treated as a single feature and thus dimensionality is drastically reduced and improved the classification accuracy.

REFERENCES

[1] Almuallim H. and Dietterich T.G., Algorithms for Identifying Relevant Features, In Proceedings of the 9th Canadian Conference on AI, pp 38-45, 1992.

[2] Almuallim H. and Dietterich T.G., Learning boolean concepts in the presence of many irrelevant features, Artificial Intelligence, 69(1-2), pp 279- 305, 1994.

[3] Arauzo-Azofra A., Benitez J.M. and Castro J.L., A feature set measure based on relief, In Proceedings of the fifth international conference on Recent Advances in Soft Computing, pp 104-109, 2004.

[4] Baker L.D. and McCallum A.K., Distributional clustering of words for text classification, In Proceedings of the 21st Annual international ACM SIGIR Conference on Research and Development in information Retrieval, pp 96- 103, 1998.

[5] Battiti R., Using mutual information for selecting features in supervised neural net learning, IEEE Transactions on Neural Networks, 5(4), pp 537- 550, 1994.

[6] Bell D.A. and Wang, H., formalism for relevance and its application in feature subset

selection, Machine Learning, 41(2), pp 175-195, 2000.

[7] Biesiada J. and Duch W., Features election for high-dimensionaldatała Pearson redundancy based filter, AdvancesinSoftComputing, 45, pp 242C249, 2008.

[8] Butterworth R., Piatetsky-Shapiro G. and Simovici D.A., On Feature Selection through Clustering, In Proceedings of the Fifth IEEE international Conference on Data Mining, pp 581-584, 2005.

[9] Cardie, C., Using decision trees to improve case-based learning, In Proceedings of Tenth International Conference on Machine Learning, pp 25-32, 1993.

[10] Chanda P., Cho Y., Zhang A. and Ramanathan M., Mining of Attribute Interactions Using Information Theoretic Metrics, In Proceedings of IEEE international Conference on Data Mining Workshops, pp 350-355, 2009.

[11] Chikhi S. and Benhammada S., ReliefMSS: a variation on a feature ranking ReliefF algorithm. Int. J. Bus. Intell. Data Min. 4(3/4), pp 375-390, 2009.

[12] Cohen W., Fast Effective Rule Induction, In Proc. 12th international Conf. Machine Learning (ICML'95), pp 115-123, 1995.

[13] Dash M. and Liu H., Feature Selection for Classification, Intelligent Data Analysis, 1(3), pp 131-156, 1997.

[14] Dash M., Liu H. and Motoda H., Consistency based feature Selection, In Proceedings of the Fourth Pacific Asia Conference on Knowledge Discovery and Data Mining, pp 98-109, 2000.

[15] Das S., Filters, wrappers and a boosting-based hybrid for feature Selection, In Proceedings of the Eighteenth International Conference on Machine Learning, pp 74-81, 2001.

[16] Dash M. and Liu H., Consistency-based search in feature selection. Artificial Intelligence, 151(1-2), pp 155-176, 2003.

[17] Demsar J., Statistical comparison of classifiers over multiple data sets, J. Mach. Learn. Res., 7, pp 1-30, 2006.

[18] Dhillon I.S., Mallela S. and Kumar R., A divisive information theoretic feature clustering algorithm for text classification, J. Mach. Learn. Res., 3, pp 1265-1287, 2003.

[19] Dougherty, E. R., Small sample issues for microarray-based classification. Comparative and Functional Genomics, 2(1), pp 28-34, 2001.

[20] Fayyad U. and Irani K., Multi-interval discretization of continuous-valued attributes for classification learning, In Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence, pp 1022-1027, 1993.